Attitude Control of Quadrotor Using PD Plus Feedforward Controller on SO(3)

Almido H Ginting[1], Oyas Wahyunggoro[2], and Adha Imam Cahyadi[3]

[1,2,3]Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada, Yogyakarta, Indonesia
[2]Department of Electrical Engineering, Universitas Nusa Cendana, Kupang, Indonesia

Article Info	ABSTRACT

This paper proposes a simple scheme of Proportional-Derivative (PD) plus Feedforward controller on SO(3) to control the attitude of a quadrotor. This controller only needs the measurement of angular velocity to calculate the exponential coordinates of the rotation matrix. With rotation matrix as an error variable of the controller, the simulation shows that the controller is able to drive the attitude of the quadrotor from hovering condition to desired attitude and from an attitude condition goes to the hovering condition, despite the system is disturbed. When the system is convergent, the rotation error matrix will be a 3×3 identity matrix.

Corresponding Author:
Adha Imam Cahyadi
Department of Electrical Engineering and Information Technology
Universitas Gadjah Mada
Yogyakarta Indonesia
Email:adha.imam@ugm.ac.id

1. INTRODUCTION

A quadrotor is a six DOF Unmanned Aerial Vehicle (UAV) which consists of two pairs of counter-rotating rotors. Due to some unique abilities of quadrotor such as high maneuverability (hovering and VTOL), small size, and easy to control, quadrotor has been widely used. The usage area of a quadrotor can be separated into three major parts such as: military operations (security intelligence), public application (SAR), and civil application (photography). In order to improve the ability of quadrotor, some researchers were carry out on several aspect of control, such as path planning [1], position [2], altitude or hovering [3], and attitude [4].

Attitude of quadrotor is an orientation of the body fixed frame with respect to the inertial frame. It depends on the moment and the thrust of the two pairs of motor. By varying the rotor speed, one can change the lift force to create motion. The ability of quadrotor depends on an attitude control of quadrotor. If the attitude control is correct, then the additional task can be added to the system, for example, in the navigation task such as path planning, the orientation of quadrotor is an important to set the accuracy of sensor to detect the obstacle. The set of attitudes of quadrotor (rigid body) is the set of 3×3 orthogonal matrix whose determinant is one as known as Special Orthogonal $SO(3)$[5].

To control the attitude of quadrotor, many of control algorithms have been developed [6]. Most of them were developed with euler angles representation, which suffer from the problem of singularities for large angle rotational maneuvers[7] [8] [9] [10][11] [12]. To avoid singularity in euler angles representation, some works use a quaternion method to represent the attitude of quadrotor [13] [14] [15]. Although the quaternions do not increase to singularities, they have double cover of the set of attitudes $SO(3)$ (no unique representation called ambiguities) in the sense that each attitude corresponds to two different quaternion vectors. Another work using a combination of Euler angles and unit-quaternion to represent the attitude of the aircraft [16].

One method that can avoid from singularities and ambiguities is a geometric control method, which uses a rotation matrix representation. This representation is global and unique. In geometric control method, the rotation matrix can be developed with an exponential matrix of exponential coordinate of an attitude, which map a vector $\Re^3$

into a matrix $SO(3)$ (Special orthogonal-3) by summing an infinite series of exponential skew-symmetric matrix. The exponential coordinate is one of the intrinsic properties of Lie group $SO(3)$.

Several studies to control a quadrotor based on exponential coordinate have been already done. In [17], a geometric tracking control system on $SO(3)$ was provided to avoid singularities and ambiguities that are inherent to other attitude representation. This proposed tracking control also show its performance to stabilize the attitude error without the knowledge of the inertia matrix. In [18], a robust adaptive tracking control of the attitude dynamics of a rigid body was proposed. The controller made a tracking for an attitude and angular velocity command without knowledge of the inertia matrix of a rigid body. Both of these researches merely give the attitude tracking approach without considering the control torque.

In [19], the attitude controller for a quadrotor based on exponential coordinate had been already done. The controller has been designed using TLC (trajectory linearisation control). The controller use an exponential coordinates $\zeta \in \Re^3$ and angular velocity $\omega \in \Re^3$ as an error variable for controller.

While in [19] uses a vector $\Re^3$ as an error variable, a rotation matrix error($\Re^{3 \times 3}$) as an error variable is used as an input of the controller in [20]. In [20], a PID algorithm is used and a knowledge of angular velocity error as an input to controller is needed. This work is look like focus in response time of the rotor.

In this paper, a simple PD plus Feedforward controller on SO(3) with a rotation matrix $R \in SO(3)$ as an error variable for the system is proposed . This controller is simpler than the controller in [20]. This controller uses a proportional-derivative algorithm and only requires the measurements of angular velocity from dynamic of quadrotor to calculate the attitude. This controller can stabilize the attitude of quadrotor in hovering condition and making a motion to a desired attitude.

2. RESEARCH METHOD

In this section, some mathematical tools will be given for rotation parametrization and some basic knowledge of Lie Group and Lie Algebra.

2.1. Mathematics of Rotation in SO(3)

Rotation matrix is a 3×3 matrix that represents the transformation vector from a fixed-body frame to an inertial frame. Rotation matrix $R \in SO(3)$ is a Lie group which has an algebraic group structure based on matrix multiplication as the group operation. According to the Euler theorem for rotates body, any 3-dimensional rotation of a rigid body can be represented by a rotation of a given axis by an angle. Suppose that $\omega \in \Re^3$ is a unit vector that specifying the rotation axis of a rigid body, and $\theta \in \Re$ is a rotation angle in radians. The position of a point at the rigid body as a function of time is denoted by $q(t)$. If the body is rotated at constant unit velocity about the axis ω, the velocity of the point can be written as

$$\dot{q}(t) = \omega \times q(t) = \hat{\omega} \, q(t). \tag{1}$$

The hat operator $^\wedge$ maps a vector in $\Re^3$ to a skew-symmetric matrix, defined as

$$\hat{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \tag{2}$$

The inverse operation of the hat operator $^\wedge$ is denoted as a vee operator $^\vee$, which recover a vector $\omega \in \Re^3$ from a skew-symmetric matrix $\hat{\omega} \in so(3)$. The skew-symmetric matrix $\hat{\omega}\theta$ satisfies $\hat{\omega}^T\theta = -\hat{\omega}\theta$. All such matrices forms a vector space denoted as $so(3)$:

$$so(3) = \left\{ S \in R^{3 \times 3} : S^T = -S \right\}. \tag{3}$$

The exponential map transforms a skew-symmetric matrices into an orthogonal matrices. Geometrically, the skew-symmetric matrix corresponds to the axis of rotation (via the mapping $\omega \mapsto \hat{\omega}$), and the exponential map generates the rotation corresponding to the rotation about the axis by a specified amount θ. So the exponential map $exp : so(3) \rightarrow SO(3)$ will maps a skew-symmetric matrix ($\hat{\omega}\theta$) to an orthogonal matrix ($R = e^{\hat{\omega}\theta}$), that is

$$R = \exp^{\hat{\omega}\theta} = I + \theta\hat{\omega} + \frac{\theta^2}{2!}\hat{\omega^2} + \frac{\theta^3}{3!}\hat{\omega^3} + ... , \tag{4}$$

where I is a 3×3 identity matrix, θ is a rotation angle, and ω is a rotation axis.

The rotation matrix $R \in SO(3)$ is a three-dimensional special orthogonal matrix that satisfies

$$R^T R = I = RR^T, \ \ det(R) = 1. \tag{5}$$

Equation (4) is an infinite series formula. To obtain a closed-form expression for $exp(\hat{\omega}\theta)$, by referred to Rodrigues' Formula, an efficient method for computing $exp(\hat{\omega}\theta)$ is stated as

$$R = \exp^{\hat{\omega}\theta} = I + \hat{\omega}\sin(\theta) + \hat{\omega}^2(1 - \cos\theta), \tag{6}$$

where $\omega = [\omega_1 \ \ \omega_2 \ \ \omega_3]^T \in \Re^3$, with $\|\omega\| = 1$ is the rotation axis, and θ is a rotation angle. The vector $\omega\theta \in \Re^3$ is the exponential coordinates ζ for $R \in SO(3)$, therefore, a rotation matrix can be calculated by an exponential matrix of a skew-symmetric matrix of exponential coordinate ζ,

$$R = e^{\hat{\zeta}}. \tag{7}$$

The inverse mapping from an orthogonal matrix $SO(3)$ to a skew-symmetric matrix $so(3)$ is done by the logarithmic map which is defined as

$$\log(R) = \frac{\theta}{2\sin\theta}(R - R^T) \in so(3), \tag{8}$$

where θ is $\cos^{-1}\left(\frac{tr(R)-1}{2}\right)$ and $|\theta| < \pi$, and $tr(R)$ is the sum of the elements on the main diagonal of rotation matrix. If $R = I$, the rotation axis ω can be chosen arbitrarily. And if $R \neq I$, the rotation axis can be calculated by $log(R)^\vee$, that is

$$\log(R)^\vee = \frac{\theta}{2\sin\theta}(R - R^T) \in \Re^3. \tag{9}$$

An illustration of a rotation with an axis and angle of rotation can be seen in Figure 1, where $p = [p_1, p_2, p_3]^T$ is a reference configuration, and $t = [t_1, t_2, t_3]^T$ is a present configuration. From Figure 1, a reference configuration p will be rotated by an angle θ to a present configuration t with a rotation axis r.
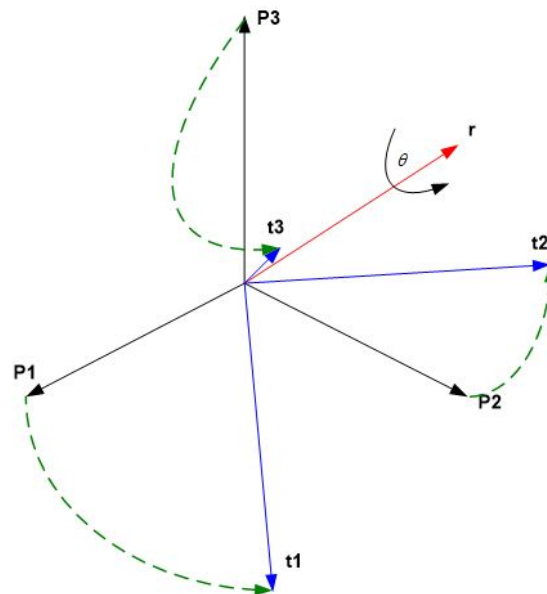

Figure 1. Illustration of rotation with axis-angle
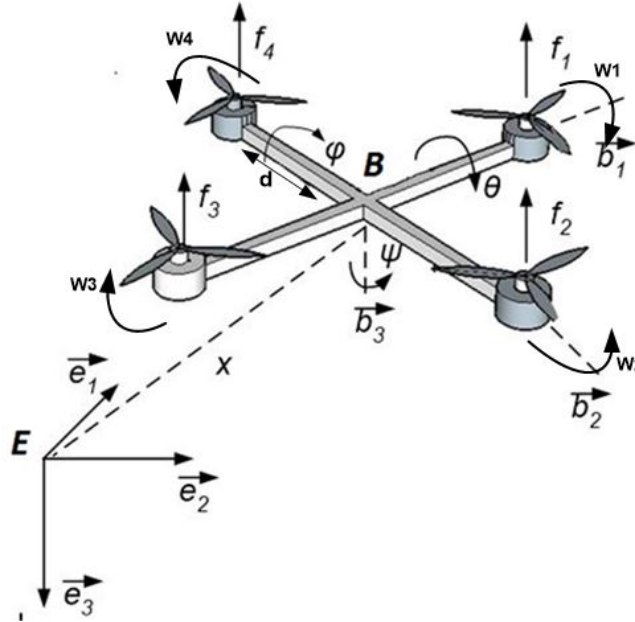
2.2. Quadrotor Model

Figure 2. Model of quadrotor

As shown in Figure 2, the inertial frame and body frame are denoted as E and B. The base of frame B can be represented using the axes of frame E, with coordinates $x_{eb}, y_{eb}, z_{eb} \in \Re^3$, where $\Re^3$ is a 3x1 dimensional-vector. In Figure 2, the quadrotor's attitude is described using roll (θ), pitch (ϕ), and yaw (ψ) angle, written as $\Psi = [\theta \ \phi \ \psi]^T$. The angular velocity is described in body frame axes, denoted as $\Omega_b = [p \ q \ r]^T$.

The inertia matrix of the quadrotor is as follows:

$$J = \begin{bmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{xy} & J_{yy} & J_{yz} \\ J_{xz} & J_{yz} & J_{zz} \end{bmatrix}, \tag{10}$$

where the indices x, y, and z denote $x-$, $y-$, and $z-$ axis in the body frame, respectively. If quadrotor is assumed to be symmetric to its $x-$ and $y-$ axis, the inertia matrix in (10) can be written as

$$J = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix}. \tag{11}$$

The inputs of quadrotor's system are the squared angular speeds of its four rotors, denoted by $W = [w_1^2 \ w_2^2 \ w_3^2 \ w_4^2]^T$. The rotors generate the torque $\tau = [\tau_x \ \tau_y \ \tau_z]^T$ and thrus T. The torques and thrust are generated by manipulating the angular speed as

$$\begin{aligned} \tau_x &= db(w_4^2 - w_2^2) \\ \tau_y &= db(w_1^2 - w_3^2) \\ \tau_z &= k(w_1^2 - w_2^2 + w_3^2 - w_4^2) \\ T &= k(w_1^2 + w_2^2 + w_3^2 + w_4^2), \end{aligned} \tag{12}$$

where d, b, k are respectively the length of the quadrotor's arm, propeller thrust coefficient, and propeller torque coefficient.

The equation of motion of this quadrotor are given by

$$\dot{R} = R\hat{\omega}^b, \tag{13}$$

$$J\dot{\omega}^b + \omega^b \times J\omega^b = \tau, \tag{14}$$

where $J \in R^{3\times 3}$, is the inertia matrix in the body fixed frame; $\omega^b \in R^3$, is the angular velocity in the body frame; $R \in (SO(3)$, is the rotation matrix as a transform from the body frame to the inertial frame; $\tau \in R^3$, is the control torque generated by the actuator of the robot.

The dynamics of quadrotor in (14) can be rewritten as

$$\dot{\omega}^b = -J^{-1}(\omega^b \times J\omega^b) + J^{-1}\tau, \tag{15}$$

and the quadrotor's kinematics in (13), referred from [21], this differential equation can be rewritten in terms of the angle ζ as

$$\dot{\zeta} = \left(I + \frac{1}{2}\hat{\zeta} + (1 - \alpha(\|\zeta\|))\frac{\hat{\zeta}^2}{\|\zeta\|^2}\right)\omega^b, \tag{16}$$

where $\alpha(\zeta) \equiv (\zeta/2)cot(\zeta/2)$, $\omega^b \in \Re^3$ is a body angular velocity, and $\zeta \in so(3)$ is an exponential coordinate of rotation matrix. Therefore, the second order system of quadrotor UAV in $SO(3)$ is defined as

$$\dot{\zeta} = \left(I + \frac{1}{2}\hat{\zeta} + (1 - \alpha(\|\zeta\|))\frac{\hat{\zeta}^2}{\|\zeta\|^2}\right)\omega^b$$
$$\dot{\omega}^b = -J^{-1}(\omega^b \times J\omega^b) + J^{-1}\tau. \tag{17}$$

2.3. PD Plus Feed-forward Control on SO(3)

Based on the works of Bullo in [22], for a simple first order system on $SO(3)$, a dynamical system with state $g \in G$ evolves following

$$\dot{g} = gV^b, \tag{18}$$

where G is a Lie Group matrix, $g \in SO(3)$ is the configuration of system, and V^b is the body velocity. Assume that the quantity of $V^b \in so(3)$ can be directly controlled to any desired value (i.e. the system is fully actuated). Then the proportional control action is

$$V^b = -k_p log(g), \quad k_p > 0. \tag{19}$$

Now, consider the stabilization problem for second order systems, that is for the system where have full control over forces (accelerations) rather than velocities. A second order system on $SO(3)$ has the form

$$\dot{g} = gV^b$$
$$\dot{V}^b = f(g, V^b) + U, \tag{20}$$

where $g \in SO(3)$ is the configuration of the system, $f(g, V^b) \in so(3)$ is the internal drift, and $U \in so(3)$ is the control input. To regulate the configuration g to the identity matrix $I \in SO(3)$, the proportional action will be coupled with a derivative term, i.e. with a term proportional to the velocity V^b. Let k_p and k_d be symmetric, positive definite gains. Then the control law is

$$U = -f(g, V^b) - k_p \log(g) - k_d V^b, \tag{21}$$

exponentially stabilizes the state g at $I \in SO(3)$ from any initial condition $tr(g(0)) - 1$ and for all k_p and $V^b(0)$ such that

$$\lambda_{min}(k_p) > \frac{\|V^b(0)\|^2}{\pi^2 - \|g(0)\|^2_{SO(3)}}, \tag{22}$$

where $\lambda_{min}(k_p)$ is the minimum eigenvalue of k_p.

Now, if the system is assumed as a quadrotor. And than, from the formulation above, if $g \in SO(3)$ is replaced with $R \in SO(3)$, V^b with ω^b, and $f(g, V^b) \in so(3)$ with $f(J\omega^b, \omega^b)$, then the control law for quadrotor is

$$\tau = \omega^b \times J\omega^b - k_p \log(R) - k_d \omega^b, \tag{23}$$

where $R \in SO(3)$ is an attitude of quadrotor, ω^b is angular velocity in body frame, J is moment inertia of quadrotor and τ is the control torque.
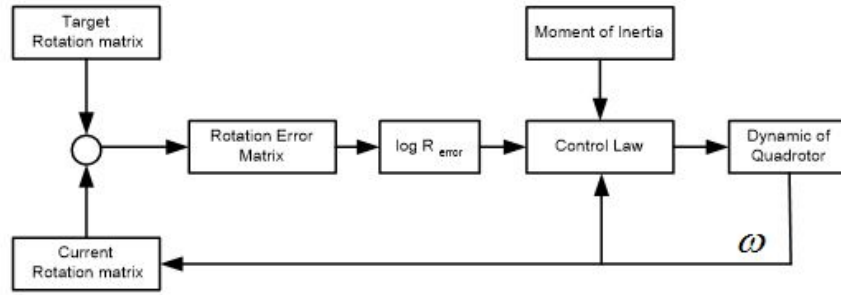
Figure 3. Block Diagram

Usually, PD control uses proportional and derivative error term. For example, if the objective is to control a variable y to match the constant set point y_d with PD control, the control law v would be

$$v = -k_p(y - y_d) - k_d \frac{d(y - y_d)}{dt}$$
$$= -k_p(y - y_d) - k_d\dot{y},$$
(24)

where k_p and k_d are proportional and derivative constant, respectively. With this error term, if the $log(R)$ is exponential coordinate of a rotation matrix ζ, the Equation (24) should be written as

$$\tau = \omega^b \times J\omega^b - k_p(\zeta - \zeta_d) - k_d\omega^b.$$
(25)

The block diagram of this controller is shown in Figure 3. From the block diagram of controller, the difference between rotation target matrix and current rotation matrix will be calculated in multiplication of matrix to get an error rotation matrix. With this multiply operation, the result is still in the form of a rotation matrix, which satisfy $R^T R = I = RR^T$ and $det(R) = 1$.

The rotation current matrix is calculated with the given angular velocity ω from dynamic model. From [21], given body angular velocity ω, the $log(R)$ which consist of exponential coordinate of rotation matrix ζ can be calculated using (7).

3. RESULT AND ANALYSIS

Here, the controller was simulated in 3 cases. In the first case, the controller is used to drive the attitude from an attitude condition to hovering condition. In the second case, the controller is used to control the attitude from hovering condition to a desired attitude. And in the last case, the simulation shows the ability of the controller to compensate an attitude disturbance.

The initial and target of rotation matrix can be an any 3×3 matrix, where the matrix satisfies the condition in (5). The controller was simulated in 6 DOF euler angles with model parameter of quadrotor are created by Corke in [23], i.e. $m = 4.34Kg$, $J_{xx} = 0.082kg.m^2$, $J_{yy} = 0.0845kg.m^2$, $J_{zz} = 0.1377kg.m^2$, $b = 1.2953 \times 10^{-5}kg.m$, $d = 0.165m$, $k = 1.0368 \times 10^{-7}kg.m^2$.

3.1. Case 1

In this case, the simulation will show the performance of controller to drive an attitude toward a hovering condition ($R1_d$). In hovering condition, $R1_d$ will be a 3×3 identity matrix. In this case, the initial attitude of quadrotor R_{init} is

$$R_{init} = \begin{bmatrix} 0.9956 & 0.06545 & -0.06767 \\ -0.06767 & 0.9972 & -0.03105 \\ 0.06545 & 0.0355 & 0.9972 \end{bmatrix}.$$

The error of attitude R_{error} can be calculated by multiplying the transpose matrix of desired attitude $R1_d$ with the initial attitude R_{init} of quadrotor, that is

$$R_{error} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T \begin{bmatrix} 0.9956 & 0.06545 & -0.06767 \\ -0.06767 & 0.9972 & -0.03105 \\ 0.06545 & 0.0355 & 0.9972 \end{bmatrix} = \begin{bmatrix} 0.9956 & 0.06545 & -0.06767 \\ -0.06767 & 0.9972 & -0.03105 \\ 0.06545 & 0.0355 & 0.9972 \end{bmatrix}.$$

The exponential coordinates of the error of attitude become an input to the controller, to calculate the required torque for the system to achieve the desired attitude. The result of the simulation is shown in Figure 4.
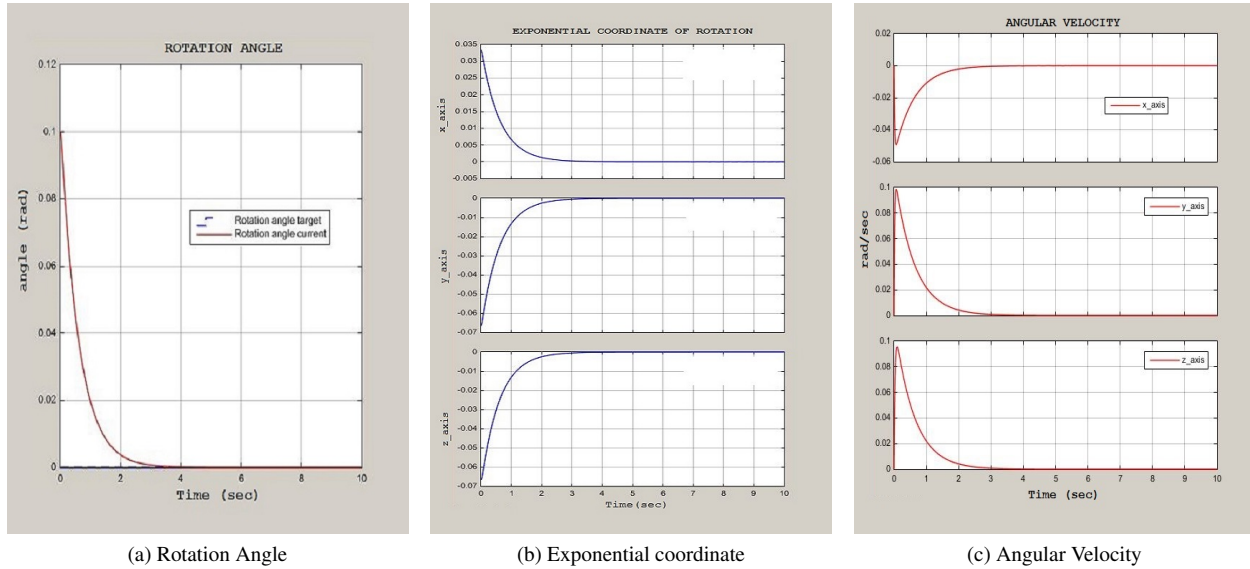

(a) Rotation Angle   (b) Exponential coordinate   (c) Angular Velocity

Figure 4. Simulation result of case 1

In Figure 4b, the exponential coordinate of this case becomes $[0\ 0\ 0]^T$ when the system is convergent. This exponential coordinates vector will be transformed into a skew-symmetric matrix by operator $^\wedge$, and the attitude of this condition can be calculated by exponential matrix of the skew-symmetric matrix of exponential coordinates as shown in (7).

$$R1_d = exp \begin{bmatrix} 0 & -0 & 0 \\ 0 & 0 & -0 \\ -0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The condition describes that the quadrotor has no rotation or in hovering condition, the same as the rotation target in this case. The rotation angle is 0 rad as shown in Figure 4a. In this condition, while the attitude $R = I$, the rotation axis ω can be chosen arbitrarily. When the system meets convergence, the angular velocity for all axis are $[0\ 0\ 0]^T$ as shown in Figure 4c. The steady state in this system is reached in 3 sec with the rise time is about 1 sec.

3.2. Case 2

In the second case, the simulation will show the performance of controller to control the quadrotors attitude from hovering condition to a desired attitude. In this case, the initial attitude R_{init} is a 3×3 identity matrix and the desired attitude $R2_d$ is

$$R2_d = \begin{bmatrix} 0.36 & 0.48 & -0.8 \\ -0.8 & 0.6 & 0 \\ 0.48 & 0.64 & 0.60 \end{bmatrix}.$$

The error of attitude R_{error} in this second case can be calculated by multiplying the transpose matrix of desired attitude $R2_d$ with the initial attitude R_{init} of quadrotor , that is

$$R_{error} = \begin{bmatrix} 0.36 & 0.48 & -0.8 \\ -0.8 & 0.6 & 0 \\ 0.48 & 0.64 & 0.60 \end{bmatrix}^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.36 & -0.80 & 0.48 \\ 0.48 & 0.60 & 0.64 \\ -0.80 & 0 & 0.6 \end{bmatrix}.$$

The result of the second case is shown in Figure 5. When the system is convergent, the value of exponential coordinate are $[0.429\ -0.858\ -0.858]^T$ (see Figure 5b), and the attitude can be calculated by the same way as shown in Case 1, that is

$$R2_d = exp \begin{bmatrix} 0 & 0.858 & -0.858 \\ -0.858 & 0 & -0.429 \\ 0.8580 & 0.429 & 0 \end{bmatrix} = \begin{bmatrix} 0.36 & 0.48 & -0.8 \\ -0.8 & 0.6 & 0 \\ 0.48 & 0.64 & 0.6 \end{bmatrix}.$$
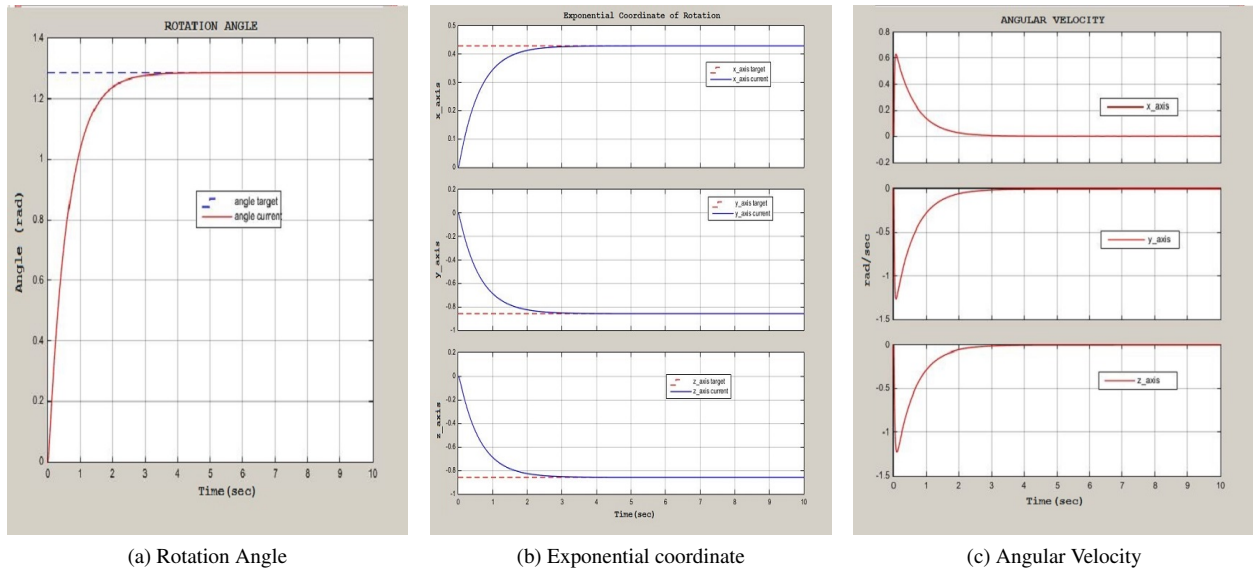

(a) Rotation Angle   (b) Exponential coordinate   (c) Angular Velocity

Figure 5. Simulation Result of case 2

As shown in Figure 5a and the value of $R2_d$, the rotation axis can be calculated by (9), that is $[0.3333 \ -0.6667 \ -0.6667]^T$. This means that the quadrotor is rotated with this rotation axis, and rotates about 1.287 rad (73.4 degree). The value of angular velocity in Figure 5c indicates that there is no alteration of rotation angle when the system is convergent.

3.3. Case 3

In the third case, the simulation is done to test the controller's ability to compensate the disturbance of the system. The controller is simulated with disturbance by $[0.1 \ 0.1 \ 0.1]^T$ in attitude current R_c, with the initial and target of attitude is the same as in Case 2. In this case, this controller can preserve the value of attitude current in 3.03 sec as shown in Figure 6a. When a disturbance is added to the system, the rotation error raises by 0.06 point in x-axis and the controller drives the system to make the error of attitude become zero in 3.03 sec as shown in Figure 6b. As long as the system is added by a disturbance, the controller preserves error in zero point. When the disturbance is eliminated from the system, rotation error in this system is reduced from zero point to point -0.06. In about 3 sec, the controller can drive the the system to the desired attitude, and the rotation error becomes zero when the system meets convergence.
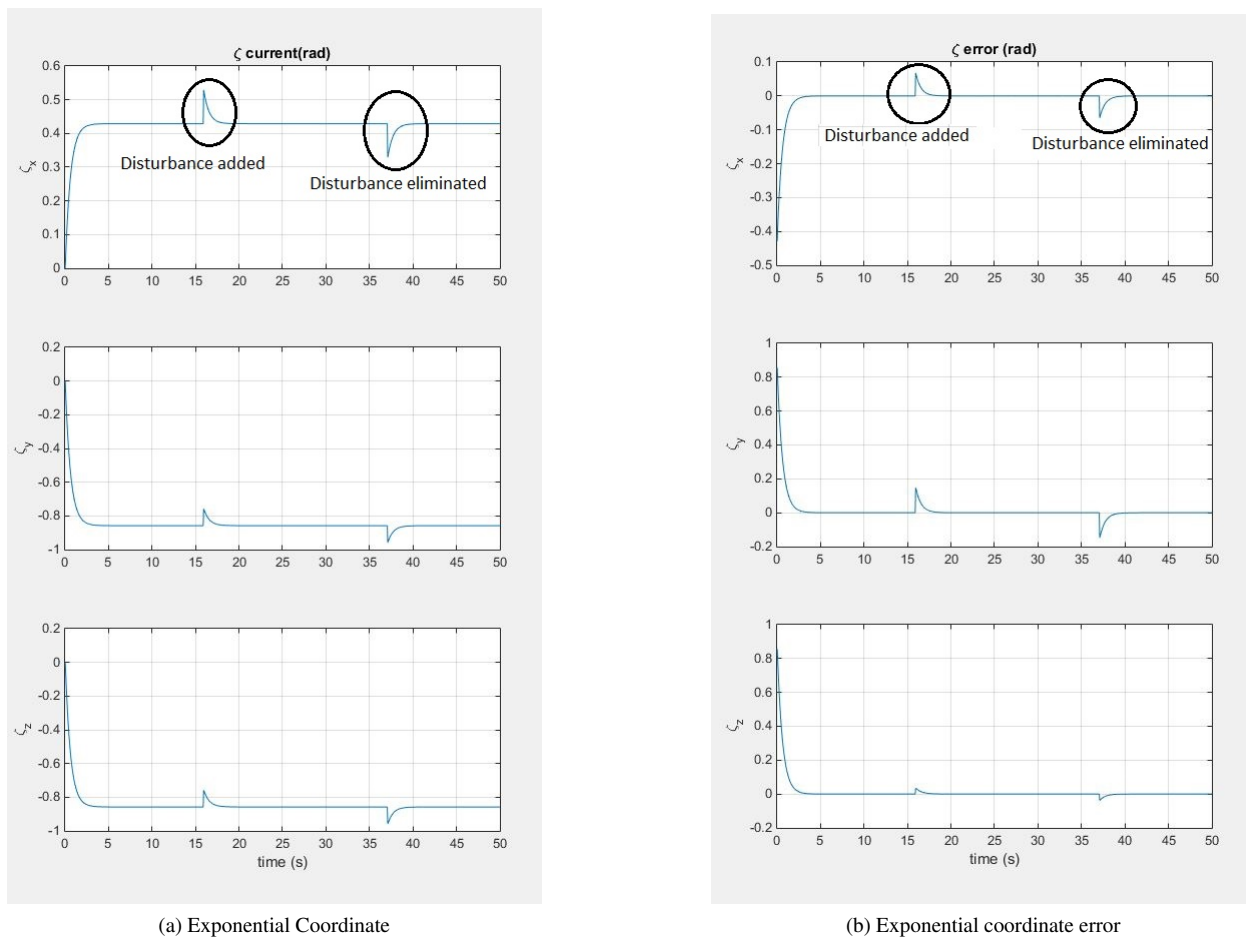
(a) Exponential Coordinate

(b) Exponential coordinate error

Figure 6. Simulation Result of case 3

Based on all of the cases, rotation error matrix is a 3×3 identity matrix when the system is convergent. The steady state time is about 3 sec and the rise time is about 1 sec for all cases.

4. CONCLUSION

A new simple scheme of PD plus Feedforward controller on SO(3) to control the attitude of quadrotor is proposed in this paper. This controller uses a rotation matrix error as a variable error of the system. The controller is able to drive the attitude of quadrotor from hovering condition to desired attitude and from an attitude condition goes to the hovering condition, with the convergence time of about 3 sec. The value of angular velocity will be zero when the desired equilibrium is achieved and the rotation error matrix will be a 3×3 identity matrix. This controller can also drive the system to the desired attitude despite the system is disturbed. For the future work, this controller will be implemented to control the maneuvering of quadrotor in the real quadrotor.

REFERENCES

[1] Iswanto, W. Oyas, and C. Adha Imam, "Path Planning Based on Fuzzy Decision Trees and Potential Field", *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, Feb 2016, pp. 212-222,
[2] H. Liu, J. Xi, and Y. Zhong, "Robust Motion Control of Quadrotors", *Journal Franklin Institude*, vol. 351, 2014 pp. 5494-5510
[3] K. Uk Lee, H. Sol Kim, J. Bae Park, and Y. Ho Choi, "Hovering Control of a Quadrotor", in *ICROS International Conference on Control, Automation and Systems*, 2012, pp. 162-167.
[4] S. Bouabdallah and R. Siegwart, "Full Control of a Quadrotor", in *IEEE/RSJ International Conference on Inteligent Robots and Systems*, 2007 pp. 153-158

[5] N. Chaturvedi, A. Sanyal, and N McClamroch, "Rigid-Body Attitude Control", *IEEE Control Systems Magazines*, vol. 31, 2011, pp. 30-51

[6] A. Zulu and S. Jhon, "A Review of Control Algorithms for Autonomous Quadrotors", *Open Journal of Applied Science*, vol. 4, 2014, pp. 547-556

[7] H. Bolandi, M. Rezaei, R. Mohsenipour, H. Nemati, and S. M. Smailzadeh, "Attitude Control of a Quadrotor with Optimized PID Controller", *Intelligent Control and Automation*, vol. 4, 2013, pp. 335-342

[8] M. N. Duc, T. N. Trong, and Y. S. Xuan,"The Quadrotor MAV System using PID Control", in *2015 Proc. IEEE International Conference on Mechatronics and Automation*, 2015 pp. 506-510.

[9] J. A. A. S. Somasiri, K. G. B. Gamagedara, D. H. S. Maithripala, and J. M. Berg, "Implementation of an Almost Globally Stable Intrinsic Nonlinear PID Controller for Attitude Stabilization of a Quadrotor", in *IEEE 10th International Conference on Industrial and Information Systems (ICIIS)*, 2015 pp. 425-428

[10] Z. He and L. Zhao, "A Simple Attitude Control of Quadrotor Helicopter Based on Ziegler-Nichols Rules for Tuning PD Parameters", *Hindawi The Scientific World Journal*, Vol. 2014, pp. 1-13.

[11] A. Jabar and F. M. Malik,"Sampled-Data Backstepping Control of a Quadrotor Unmanned Aerial Vehicle", *International Journal of Robotics and Automation (IJCRA)* (2015), Vol. 4, No. 2, June 2015, pp. 124-134.

[12] Z. Lian Chuan and K. To Chong, "Nonlinear Hybrid Controller for a Quadrotor Based on Sliding Mode and Backstepping", *International Journal of Robotics and Automation (IJCRA)* (2015), Vol. 4, No. 3, September 2015, pp. 209-218.

[13] A. Tayebi and S. Mcgilvary, "Attitude Stabilization of a VTOL Quadrotor Aircraft", *IEEE Transaction on Control Systems Technology*, vol.14, no. 3, 2006, pp. 562-571

[14] O. Magnussen, M. Ottestad, and G. Hovland, "Experimental validation of a quaternion-based attitude estimation with direct input to a quadcopter control system", in *International Conference on Unmanned Aircraft Systems, ICUAS*, 2013, pp. 480-485

[15] E. Fresk and G. Nikolakopoulos, "Full Quaternion Based Attitude Control for a Quadrotor", in *European Control Conference (ECC)*, 2013, pp. 3864-3869

[16] K. D. Do, "Global Tracking Control of Quadrotor VTOL Aircraft in Three-Dimensional Space", *International Journal of Robotics and Automation (IJCRA)* (2015), Vol. 3, No. 1, March 2014, pp. 13-29.

[17] T. Lee, "Geometric Tracking Control of the Attitude Dynamics of a Rigid Body on So(3)", in *2011 American Control Conference*,Jul 2011, pp. 1200-1205.

[18] T. Lee, "Robust Adaptive Attitude Tracking on SO(3) With an Application to a Quadrotor UAV", *IEEE Transactions on Control Systems Technology*, vol. 21, No. 5, pp. 1924-1930, Sept 2013

[19] Y. Yu, X. Ding, and J. J. Zhu, "Attitude Tracking Control of a Quadrotor UAV in the Exponential Coordinates", *Journal of The Franklin Institute*, vol. 350, pp. 2044-2068, 2013

[20] Y. Yu, S. Yang, M. Wang, C. Li, and Z. LI, "High Performance Full Attitude Control of a Quadrotor on SO(3)", in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 26-30, 2015, pp. 1698-1703.

[21] F. Bullo and R. M. Murray, "Proportional Derivative (PD) Control on the Euclidean Group", in *CDS Technical Report 95-010*, August 1995.

[22] F. Bullo and R. M. Murray, "Proportional Derivative (PD) Control on the Euclidean Group", in *1995 European Control Conference*, Sept 1995, pages 10911097.

[23] P. I. Corke, "Robotics, Vision and Control: Fundamental algorithms in MATLAB", Berlin, Springer, 2011.